

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-319546

(43)Date of publication of application : 12.12.1997

(51)Int.Cl. G06F 3/14

(21)Application number : 08-133893

(71)Applicant : MITSUBISHI ELECTRIC CORP

(22)Date of filing : 28.05.1996

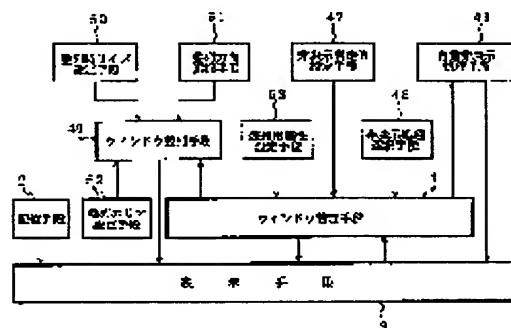
(72)Inventor : HAMAGUCHI TEI

(54) MULTIWINDOW DISPLAY DEVICE

(57)Abstract:

PROBLEM TO BE SOLVED: To reduce the operations of an operator for confirming display contents by performing non-display processing to a window when the relevant window is not utilized throughout set reference time.

SOLUTION: When the window is mapped, while referring to the number of counting for deciding the non-display of window control data, a window managing means 1 discriminates whether this window is an automatic display object or not. When the window is discriminated as a non-display object, it is reported to the window identifier, count number and automatic non-display processing means of window control data as a mapping/focus-out report. When the window is turned into non-display or focused, the count number of window control data is referred to and when that window is discriminated as a non-display object, the window identifier of the window control data is reported to an automatic non-display processing means 46 as a non-display/focus-in report.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-319546

(43) 公開日 平成9年(1997)12月12日

(51) Int. Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 3/14	3 5 0		G 0 6 F 3/14 3 5 0 A	

審査請求 未請求 請求項の数 8 O L (全 23 頁)

(21) 出願番号 特願平8-133893

(22) 出願日 平成8年(1996)5月28日

(71) 出願人 000006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(72) 発明者 濱口 稔

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

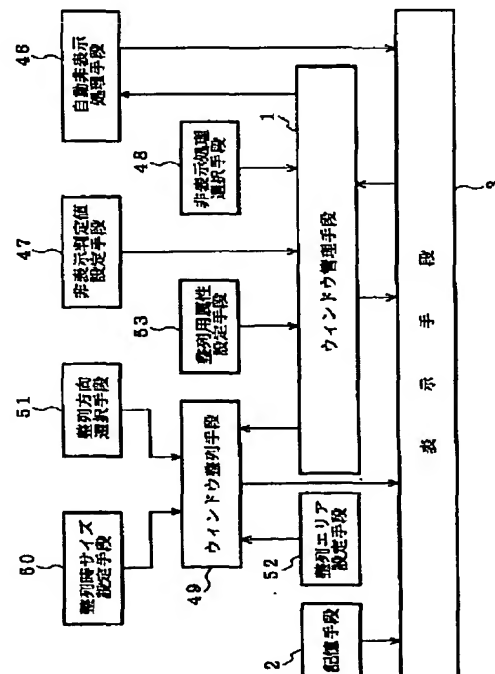
(74) 代理人 弁理士 田澤 博昭 (外2名)

(54) 【発明の名称】 マルチウィンドウ表示装置

(57) 【要約】

【課題】 不要となったウィンドウは、その都度非表示としない限りオーバーラップ状態が発生し、表示内容の確認にマウス操作によるウィンドウの表示優先順位変更等の手間がかかる。

【解決手段】 設定された基準時間を通じてウィンドウが利用されなかった場合に該当するウィンドウを非表示処理する自動非表示処理手段46を備えた。



【特許請求の範囲】

【請求項1】 ウィンドウの大きさ、位置およびウィンドウ間の表示順序をウィンドウ制御データに基づいて管理制御するウィンドウ管理手段と、上記ウィンドウの表示データを格納する記憶手段と、上記ウィンドウ管理手段に管理されているウィンドウ制御データに基づいて上記ウィンドウを表示すると共に上記記憶手段に格納されている表示データを該当するウィンドウ内に表示する表示手段と、設定された基準時間を通じてウィンドウが利用されなかった場合に該当するウィンドウを非表示処理する自動非表示処理手段とを備えたマルチウィンドウ表示装置。

【請求項2】 ウィンドウ毎に基準時間を任意に設定する非表示判定値設定手段を備えたことを特徴とする請求項1記載のマルチウィンドウ表示装置。

【請求項3】 ウィンドウ毎に非表示処理内容をアイコン化またはウィンドウ削除に任意に設定する非表示処理選択手段を備えたことを特徴とする請求項1または請求項2記載のマルチウィンドウ表示装置。

【請求項4】 ウィンドウの大きさ、位置、ウィンドウ間の表示順序、ウィンドウの整列時の識別子および整列時のウィンドウの整列順序をウィンドウ制御データに基づいて管理制御するウィンドウ管理手段と、上記ウィンドウの表示データを格納する記憶手段と、上記ウィンドウ管理手段に管理されているウィンドウ制御データのうちウィンドウの整列時の識別子および整列時のウィンドウの整列順序に応じた定義ファイルに設定された整列時のウィンドウサイズ、整列エリアの座標および整列アルゴリズムに基づいて整列時の各ウィンドウの表示位置を演算するウィンドウ整列手段と、上記ウィンドウ管理手段に管理されているウィンドウ制御データのうちウィンドウの大きさ、位置およびウィンドウ間の表示順序と上記ウィンドウ整列手段により演算された整列時の各ウィンドウの表示位置に基づいて上記ウィンドウを表示すると共に上記記憶手段に格納されている表示データを該当するウィンドウ内に表示する表示手段とを備えたマルチウィンドウ表示装置。

【請求項5】 整列時のウィンドウサイズを任意に設定する整列時サイズ設定手段を備えたことを特徴とする請求項4記載のマルチウィンドウ表示装置。

【請求項6】 整列アルゴリズムを任意に設定する整列方向選択手段を備えたことを特徴とする請求項4または請求項5記載のマルチウィンドウ表示装置。

【請求項7】 整列エリアの座標を任意に設定する整列エリア設定手段を備えたことを特徴とする請求項4から請求項6のうちのいずれか1項記載のマルチウィンドウ表示装置。

【請求項8】 ウィンドウの整列時の識別子および整列時のウィンドウの整列順序を任意に設定する整列用属性設定手段を備えたことを特徴とする請求項4から請求項

7のうちのいずれか1項記載のマルチウィンドウ表示装置。

【発明の詳細な説明】**【0001】**

【発明の属する技術分野】この発明は、複数のウィンドウを重ね合せて表示するマルチウィンドウ表示が可能なマルチウィンドウ表示装置に関するものである。

【0002】

【従来の技術】図16は例えば特開昭63-4328号公報に示された従来のマルチウィンドウ表示装置を示す機能ブロック図であり、図において、1は各ウィンドウの大きさ、位置およびウィンドウ間の表示順序を管理するウィンドウ管理手段、2は該当するウィンドウの表示データを格納する記憶手段、3は記憶手段2に格納されている各ウィンドウの表示データを取り出して、各ウィンドウの画面をウィンドウ管理手段1によって管理される属性、表示順序等のウィンドウ制御データに基づいてディスプレイ24に重ね合わせて表示する表示手段である。

【0003】また、図17はマルチウィンドウ表示装置のシステム構成図であり、図において、10はマイクロプロセッサ11の制御下で作動し、複数のウィンドウをディスプレイ24に同時に重ね合わせて表示するマルチウィンドウ表示装置である。マイクロプロセッサ11はウィンドウの生成、削除、変更を行なう。また、マイクロプロセッサ11にはプログラムの他にオペレータの操作でウィンドウへの制御情報および表示データ情報等を入力するためのマウス12およびキーボード13がマウス制御装置14およびキーボード制御装置15を介して接続されている。マイクロプロセッサ11はさらにウィンドウ管理テーブル16、ウィンドウメモリ17およびオーバーラップ制御装置18等の周辺デバイスを制御する。

【0004】ウィンドウ管理テーブル16はウィンドウの大きさ、位置およびウィンドウ間の表示順序関係を管理するテーブルであり、個々のウィンドウがこのウィンドウ管理テーブル16を持ち、マイクロプロセッサ11によって生成、削除、変更が行なわれる。ウィンドウメモリ17はウィンドウの表示データを格納するメモリで各ウィンドウに対応した論理バッファが用意されている。オーバーラップ制御装置18はマイクロプロセッサ11の制御下でウィンドウ管理テーブル16から各ウィンドウの表示制御情報を取り出して画面に表示すべき制御データを表示制御メモリ19に格納するものである。

【0005】また、ウィンドウ表示装置20はオーバーラップ制御装置18が表示制御メモリ19に制御データを格納した後、表示制御メモリ19に格納された制御データを取りだし、ビットマップ21に出力するものである。このビットマップ21は、ウィンドウの表示データをウィンドウメモリ17から取りだし、この表示データ

とウィンドウ表示装置 20 からの制御データとを表示メモリ 22 に格納するものである。表示制御装置 23 はディスプレイ 24 の表示を制御し、表示メモリ 22 に格納されたウィンドウ毎の表示データを制御データに基づいてディスプレイ 24 に表示するものである。ディスプレイ 24 は表示制御装置 23 の制御下で複数のウィンドウを同時にオーバーラップした状態で表示することができるようになっている。

【0006】図 18 はウィンドウ管理テーブル 16 に生成される格納情報を示す概念図であり、ウィンドウ管理テーブル 16 には、ウィンドウの表示データ格納用論理バッファ LBF の幅 (LBW) 31、高さ (LBH) 32、ウィンドウの幅 (WIW) 33 と高さ (WIH) 34、ウィンドウの画面上の X 座標 (WIX) 35、Y 座標 (WIY) 36、次のウィンドウへのポインタ (FPA) 37、前のウィンドウへのポインタ (BPA) 38、ウィンドウの前景色 (FGC) 39 および背景色 (BGC) 40 等のウィンドウ制御データ 30 が格納される。

【0007】次に動作について説明する。図 19

(a), (b), (c) はマイクロプロセッサにより 2 つのウィンドウ (WINA), (WINB) を生成した時のウィンドウメモリ 17 およびウィンドウ管理テーブル 16 の状態を示す説明図である。図 19 (a) に示すようにウィンドウメモリ 17 にはウィンドウ (WINA) 用の論理バッファ (LBFA) とウィンドウ (WINB) 用の論理バッファ (LBFB) がマイクロプロセッサ 11 のプログラムにより生成され、それぞれ (WINA), (WINB) の表示データが格納されている。

【0008】図 19 (c) に示すように、ディスプレイ 24 にウィンドウ (WINA) を最前面にウィンドウ (WINB) を最後面に表示するにはウィンドウ管理テーブル 16 のリンク構造が図 19 (b) に示すようになっていなければならない。即ち、ウィンドウ管理テーブル 16 のベーステーブル (BTBL) のベースポインタ (BSP) は最前面のウィンドウ管理テーブル 16 の先頭アドレスをポイントするもので、今の場合、ベースポインタ (BSP) はウィンドウ (WINA) の管理テーブル (MTBA) の先頭アドレスをポイントしている。また、ウィンドウ (WINA) の管理テーブル (MTBA) のポインタ (BPA) は、今の場合、ウィンドウ (WINA) は最前面であり、このウィンドウ (WINA) よりも前にはウィンドウがないので終端コード (NULL) が格納されている。

【0009】一方、管理テーブル (MTBA) のポインタ (FPA) は、今の場合、次のウィンドウ (WINB) の管理テーブル (MTBB) の先頭アドレスをポイントしている。ウィンドウ (WINB) の管理テーブル (MTBB) のポインタ (FPB) は、今の場合、ウィンドウ (WINB) は最後面であり、このウィンドウ (WINB) よりも後にはウィンドウがないので終端コ

ード (NULL) が格納されている。一方、管理テーブル (MTBB) のポインタ (BPB) は、今の場合、前のウィンドウ (WINA) の管理テーブル (MTBA) の先頭アドレスをポイントしている。

【0010】このようにして、図 19 (c) に示すように、ディスプレイ 24 の画面上にウィンドウ (WINA) をウィンドウ (WINB) の上に重ねて表示している時には、ウィンドウ (WINB) の論理バッファ (LBFB) に表示を行なった場合、オペレータがウィンドウ (WINB) の表示内容を知ることができない場合が生じる。このときは、マウス 12 またはキーボード 13 の操作あるいはこのマルチウィンドウ表示装置 10 を利用しているアプリケーションプログラムにより、ウィンドウ (WINB) が最前面になるようウィンドウの表示順序を入れ換える。

【0011】これはウィンドウ管理テーブル 16 のベーステーブル (BTBL) のポインタ (BSP) がウィンドウ (WINB) の管理テーブル (MTBB) の先頭アドレスをポイントし、ウィンドウ (WINB) の管理テーブル (MTBB) のポインタ (BPB) に終端コードの (NULL) を格納し、管理テーブル (MTBB) のポインタ (FPB) がウィンドウ (WINA) の管理テーブル (MTBA) の先頭アドレスをポイントし、ウィンドウ (WINA) の管理テーブル (MTBA) のポインタ (FPA) に終端コード (NULL) を格納し、管理テーブル (MTBA) のポインタ (BPA) がウィンドウ (WINB) の管理テーブル (MTBB) の先頭アドレスをポイントするように、マイクロプロセッサ 11 のプログラムでウィンドウ管理テーブル 16 のベーステーブル (BTBL) のポインタ (BSP) および各ウィンドウの管理テーブル (MTBA), (MTBB) のポインタ (FPA), (BPA), (FPB), (BPB) を変更することによってなされる。

【0012】

【発明が解決しようとする課題】従来のマルチウィンドウ表示装置は以上のように構成されているので、画面上に表示されているウィンドウは、オペレータまたはアプリケーション S/W が非表示とする対象の識別を含め明示的に非表示操作を行なわない限り非表示とならず、そのため必要な情報が複数のウィンドウに分散して表示される場合、不要となったウィンドウはその都度、非表示としない限りオーバーラップ状態が発生し、表示内容の確認にマウス操作によるウィンドウの表示優先順位変更等の手間がかかる課題があった。

【0013】また、上記オーバーラップ状態を解消する場合、オペレータはウィンドウの表示内容とそこから得る情報の有用性および優先度等により非表示とするウィンドウを選択しなければならず、ウィンドウの表示枚数が多い場合、非表示操作も含めオペレータに相当の負荷が発生するという課題があった。

【0014】さらに、個々のコマを1つのウィンドウに表示する動画のコマ送り表示のように、表示内容が類似したものや、関連する情報が複数のウィンドウに表示されている場合、当初ある法則に従って配置されていた表示順序を崩してしまったり、当初から何の規則性もなく配置されていた場合、オペレータは表示内容より個々のウィンドウの関連を識別し、さらには必要に応じてマウス操作等によりウィンドウの再配置を行わなければならない、オペレータに過度の負荷が発生するという課題があった。

【0015】この発明は上記のような課題を解決するためになされたもので、複数のウィンドウが表示されている場合に、一定時間利用しなかったウィンドウを識別して非表示とする操作と、表示順序に規則性がある個々のウィンドウの関係を識別し、その表示順序に合致させるための表示操作をオペレータに代わって行なうマルチウィンドウ表示装置を得ることを目的とする。

【0016】

【課題を解決するための手段】請求項1記載の発明に係るマルチウィンドウ表示装置は、設定された基準時間を通じてウィンドウが利用されなかった場合に該当するウィンドウを非表示処理する自動非表示処理手段を備えたものである。

【0017】請求項2記載の発明に係るマルチウィンドウ表示装置は、ウィンドウ毎に基準時間を任意に設定する非表示判定値設定手段を備えたものである。

【0018】請求項3記載の発明に係るマルチウィンドウ表示装置は、ウィンドウ毎に非表示処理内容をアイコン化またはウィンドウ削除に任意に設定する非表示処理選択手段を備えたものである。

【0019】請求項4記載の発明に係るマルチウィンドウ表示装置は、ウィンドウの大きさ、位置、ウィンドウ間の表示順序、ウィンドウの整列時の識別子および整列時のウィンドウの整列順序をウィンドウ制御データに基づいて管理制御するウィンドウ管理手段と、上記ウィンドウの表示データを格納する記憶手段と、上記ウィンドウ管理手段に管理されているウィンドウ制御データのうちウィンドウの整列時の識別子および整列時のウィンドウの整列順序に応じた定義ファイルに設定された整列時のウィンドウサイズ、整列エリアの座標および整列アルゴリズムに基づいて整列時の各ウィンドウの表示位置を演算するウィンドウ整列手段と、上記ウィンドウ管理手段に管理されているウィンドウ制御データのうちウィンドウの大きさ、位置およびウィンドウ間の表示順序と上記ウィンドウ整列手段により演算された整列時の各ウィンドウの表示位置に基づいて上記ウィンドウを表示すると共に上記記憶手段に格納されている表示データを該当するウィンドウ内に表示する表示手段とを備えたものである。

【0020】請求項5記載の発明に係るマルチウィンド

ウ表示装置は、整列時のウィンドウサイズを任意に設定する整列時サイズ設定手段を備えたものである。

【0021】請求項6記載の発明に係るマルチウィンドウ表示装置は、整列アルゴリズムを任意に設定する整列方向選択手段を備えたものである。

【0022】請求項7記載の発明に係るマルチウィンドウ表示装置は、整列エリアの座標を任意に設定する整列エリア設定手段を備えたものである。

【0023】請求項8記載の発明に係るマルチウィンドウ表示装置は、ウィンドウの整列時の識別子および整列時のウィンドウの整列順序を任意に設定する整列用属性設定手段を備えたものである。

【0024】

【発明の実施の形態】以下、この発明の実施の一形態を説明する。

実施の形態1. 図1はこの発明の実施の形態によるマルチウィンドウ表示装置を示す機能ブロック図であり、図において、1はウィンドウの大きさ、位置およびウィンドウ間の表示順序等を図2に示すウィンドウ制御データ30に基づいて管理制御するウィンドウ管理手段、2はウィンドウの表示データを格納する記憶手段、3は上記ウィンドウ管理手段1に管理されているウィンドウ制御データ30に基づいてウィンドウを表示すると共に上記記憶手段2に格納されている表示データを該当するウィンドウ内に表示する表示手段である。46はフォーカスの有無によりウィンドウの利用度を判定し、システムに設定されている経過時間内にフォーカスされなかったウィンドウを自動的に非表示とする自動非表示処理手段である。ウィンドウ管理手段1、記憶手段2および表示手段3は、従来の技術とほぼ同様な構成であり、ウィンドウ管理手段1には、従来の機能の他に自動非表示処理手段46に必要な情報を通知する機能が付加されている。

【0025】図2はこの発明の実施の形態によるウィンドウ管理手段1が管理するウィンドウ管理テーブルを示す概念図であり、ウィンドウ管理テーブルには、ウィンドウの表示データ格納用論理バッファLBFの幅(LBW)31、高さ(LBH)32、ウィンドウの幅(WIW)33と高さ(WIH)34、ウィンドウの画面上のX座標(WIX)35、Y座標(WIY)36、次のウィンドウへのポインタ(FP)37、前のウィンドウへのポインタ(BP)38、ウィンドウの前景色(FGC)39および背景色(BGC)40等の以上、従来の技術に示したウィンドウ制御データ30が格納され、また、ウィンドウ識別子(win_id)41、自動非表示処理手段46が非表示と判定するためのカウント数(judge)42、自動非表示の処理内容(process)43を付加したウィンドウ制御データ30が格納されている。

【0026】次に動作について説明する。ウィンドウ管理手段1はウィンドウがマッピングまたはフォーカスア

ウトされると、当該ウィンドウ制御データ30の非表示と判定するためのカウント数(judge)42を参照し、当該ウィンドウが自動非表示対象か否かを判定する。そして、当該ウィンドウが非表示対象と判定されると、マッピング/フォーカスアウト通知として当該ウィンドウ制御データ30のウィンドウ識別子(win_id)41、カウント数(judge)42および自動非表示の処理内容(process)43を自動非表示処理手段46に通知する。また、ウィンドウが非表示となったり、フォーカスされると当該ウィンドウ制御データ30のカウント数(judge)42を参照し、当該ウィンドウが自動非表示対象か否かを判定する。そして、当該ウィンドウが非表示対象と判定されると、非表示/フォーカスイン通知として当該ウィンドウ制御データ30のウィンドウ識別子(win_id)41を自動非表示処理手段46に通知する。

【0027】さらに、図3(a)、(b)は自動非表示処理手段の動作を示すフローチャート、図4(a)、

(b)は自動非表示処理手段が管理する非表示ウィンドウデータ、および非表示対象ウィンドウリストを示す概念図であり、この図3および図4を用いて自動非表示処理手段46の処理概要を説明する。初期化処理(ステップST1)後に、自動非表示処理手段46はウィンドウ管理手段1よりマッピング/フォーカスアウト通知を受信すると(ステップST2)、図4(a)に示す非表示ウィンドウデータ55を生成し、図4(b)に示す非表示対象ウィンドウリスト62の終端に生成した非表示ウィンドウデータ55を登録する(ステップST3)。

【0028】また、ウィンドウ管理手段1より非表示/フォーカスイン通知を受信すると(ステップST4)、当該ウィンドウの非表示ウィンドウデータ55を非表示対象ウィンドウリスト62より削除する(ステップST5)。上記ウィンドウ管理手段1からの通知チェックおよび通知があった場合の非表示対象ウィンドウリスト62へのデータの登録/削除処理を終えると(ステップST6)、自動非表示処理手段46は非表示対象ウィンドウリスト62に登録されている非表示ウィンドウデータ55のフォーカスアウトされてからの経過カウント値(counter)59(初期値0)を1カウントアップし(ステップST7)、経過カウント値(counter)59の値が非表示と判定されるカウント値(judgment)58と一致したウィンドウを非表示処理の内容を表すフラグ(unmap_flag)60に設定されている非表示処理にて非表示とする(ステップST8、ステップST9)。さらに、次管理データへのポインタ(next_tabl)61が終端コード(NULL)であるか判定し(ステップST10)、終端コード(NULL)である場合はステップST2に戻り、終端コード(NULL)でない場合は次管理データへのポインタに進む(ステップST11)。さらに、図3

(b)において、非表示処理(ステップST9)は、非表示処理の内容を示すundisp_flagを判定し(ステップST91)、ウィンドウ削除処理(ステップST92)かアイコン化処理(ステップST93)を行い、管理テーブルからデータを削除して(ステップST94)、処理を終了する。

【0029】図5は非表示ウィンドウデータの遷移を示す概念図であり、この図5を用いて自動非表示処理手段46におけるウィンドウの利用度の識別ロジックを説明する。図5(a)は最初に非表示対象外のウィンドウ

(A)がマッピングされ、その後に非表示対象であるウィンドウ(B)(非表示判定値42:180ループ、非表示処理43:アイコン化)がマッピングされ、上記カウントアップ処理が既に155ループ行なわれた場合の非表示ウィンドウデータ55および非表示対象ウィンドウリスト62を示したものである。図5(a)においてウィンドウ(A)は非表示対象外であるため、自動非表示処理手段46により非表示ウィンドウデータ55は非表示対象ウィンドウリスト62に登録されておらず、非表示対象であるウィンドウ(B)の非表示ウィンドウデータ55だけが登録され、ウィンドウ識別子(window_id)56にはB、非表示と判定されるカウント値(judgment)58には180、経過カウント値(counter)59には155が設定されている。

【0030】図5(b)は図5(a)のような初期状態より、10ループの時間が経過した後に非表示対象であるウィンドウ(C)(非表示判定値42:300、非表示処理43:ウィンドウ削除)がマッピングされた場合の非表示ウィンドウデータ55および非表示対象ウィンドウリスト62を示したものである。ウィンドウ(C)は非表示対象であるため、ウィンドウ管理手段1からマッピング/フォーカス通知が通知され、通知を受けた自動非表示処理手段46は非表示対象ウィンドウリスト62の終端にウィンドウ(C)の非表示ウィンドウデータ55を登録すると共に、常時、定周期で非表示対象ウィンドウリスト62に登録されているウィンドウの経過カウント値(counter)59をカウントアップするため、ウィンドウ(B)の経過カウント値(counter)59は165となっている。

【0031】図5(c)は図5(b)の状態から、さらに、10ループの時間が経過した後に、ウィンドウ

(B)がフォーカスされた場合の非表示ウィンドウデータ55を示したものである。ウィンドウ(B)は非表示対象であるため、ウィンドウ管理手段1から非表示/フォーカスイン通知が通知され、通知を受けた自動非表示処理手段46は非表示対象ウィンドウリスト62を検索し、ウィンドウ(B)の非表示ウィンドウデータ55を非表示対象ウィンドウリスト62より削除する。そのためウィンドウ(B)はフォーカスされている間、カウ

トアップの対象外となり、フォーカスアウトされた時点で、非表示対象ウィンドウリスト62に非表示ウィンドウデータ55が登録され、ウィンドウ(B)の経過カウンタ値(counter)59は初期値にリセットされる。

【0032】以上のように、この実施の形態1によれば、自動非表示処理手段46はフォーカスの有無により、非表示ウィンドウデータ55内の経過カウンタ値(counter)59をカウンタアップおよびリセットすることで、オペレータに代わりウィンドウの利用度を識別できるので、利用度の低いウィンドウを自動的に非表示とすることで、表示内容確認のためのオペレータの操作を軽減できると共に、同時に監視画面の視認性および識別性を向上することができる効果が得られる。

【0033】実施の形態2。図1において、47はウィンドウ管理手段1内に設けられ、実施の形態1ではアプリケーションS/Wがウィンドウ生成時にしか設定できなかった非表示と判定するカウンタ数(judge)42の値をオペレータがウィンドウ毎に任意の値で設定可能にする非表示判定値設定手段である。その他の構成については実施の形態1と同様なので、その重複する説明を省略する。

【0034】次に動作について説明する。図6はこの発明の実施の形態2による非表示判定値設定手段47が表示するポップアップウィンドウとその動作を示すフローチャートである。オペレータがウィンドウのタイトルバー等に設けられた非表示判定値設定用ボタンをマウスまたはタッチパネル等のポインティングデバイスにてタッチまたはクリックすると(ステップST15)、非表示判定値設定手段47は図6(a)のように新規の非表示判定値を入力するためのポップアップウィンドウを表示し(ステップST16)、オペレータの入力を促すと共にウィンドウ管理テーブルを検索し、当該カウンタ数(judge)42を入手する。オペレータがポップアップウィンドウを閉じると、非表示判定値設定手段47は入手したカウンタ数(judge)42の値と入力された非表示判定値とを比較し、一致しなかった場合だけ(ステップST17)、当該ウィンドウ制御データ30のカウンタ数(judge)42を更新し、ウィンドウ管理手段1の非表示判定値変更通知として当該ウィンドウ制御データ30のウィンドウ識別子(win_id)41、カウンタ数(judge)42を自動非表示処理手段46に通知する(ステップST18)。また、上記比較の際に、何も入力されていなかった、またはカウンタ数(judge)42の値に変更がない場合はそのまま処理を終了する。

【0035】図7はこの発明の実施の形態2における自動非表示処理手段46の動作を示すフローチャートである。実施の形態2における自動非表示処理手段46はオペレータによる非表示判定値設定のため、実施の形態1

の処理の他に、ウィンドウ管理手段1からの非表示判定値変更通知のチェックと、通知を受けた場合の非表示ウィンドウデータ55の更新処理を付加している(ステップST20～ステップST23)。

【0036】以上のように、この実施の形態2によれば、実施の形態1ではシステム稼働中に変更できなかった非表示判定値の値をシステム稼働中でも変更可能とすることで、表示内容および利用状況といった実情に応じた判定値が柔軟に設定可能となるため、容易に最適値を設定可能とする効果が得られると共にシステムとしての汎用性向上の効果も得られる。

【0037】実施の形態3。図1において、48はウィンドウ管理手段1内に設けられ、実施の形態1ではアプリケーションS/Wがウィンドウ生成時にしか設定できなかった非表示処理の内容(process)43の値をオペレータがウィンドウ毎に任意の値で設定可能とする非表示処理選択手段である。その他の構成については実施の形態1ないし2と同様なので、その重複する説明を省略する。

【0038】次に動作について説明する。図9はこの発明の実施の形態3による非表示処理選択手段48が表示するポップアップウィンドウとその動作を示すフローチャートである。オペレータがウィンドウのタイトルバー等に設けられた非表示処理選択用ボタンをマウスまたはタッチパネル等のポインティングデバイスにてタッチまたはクリックすると(ステップST35)、非表示処理選択手段48は図9(a)のように非表示処理内容を選択するためのポップアップウィンドウを表示し(ステップST36)、オペレータの入力を促すと共に、ウィンドウ管理テーブルを検索し、当該ウィンドウの非表示処理の内容(process)43を入手する。オペレータがポップアップウィンドウを閉じると、非表示処理選択手段48は入手した非表示処理の内容(process)43の値と入力された非表示処理内容とを比較し、一致しなかった場合だけ(ステップST37)、当該ウィンドウ制御データ30の非表示処理の内容(process)43を更新し、ウィンドウ管理手段1の非表示処理変更通知として当該ウィンドウ制御データ30のウィンドウ識別子(win_id)41、非表示処理の内容(process)43を自動非表示処理手段46に通知する(ステップST38)。また、上記比較の際に、何も入力されていなかった、または非表示処理の内容(process)43の値に変更がない場合は、そのまま処理を終了する。そして、実施の形態2における非表示判定値変更通知と非表示処理変更通知の通知用処理を汎用化するため通知の種類を設定する(value_kind)というエンタリを新設し、両通知の通知データフォーマットを統一する。

【0039】図8はこの発明の実施の形態3における自動非表示処理手段46の動作を示すフローチャートであ

る。実施の形態3における自動非表示処理手段46はオペレータによる非表示処理設定のため、実施の形態2の処理の他に、ウィンドウ管理手段1からの非表示処理変更通知のチェックと、通知を受けた場合の非表示ウィンドウデータ55の更新処理、また非表示判定値変更通知と非表示処理変更通知のデータフォーマット統一による通知内容識別手段を付加している（ステップST27～ステップST32）。

【0040】以上のように、この実施の形態3によれば、実施の形態1ないし2ではシステム稼働中に変更できなかった非表示処理内容の値をシステム稼働中でも変更可能とすることで、表示内容および利用状況といった実情に応じた非表示処理が柔軟に設定可能となるため、最適な非表示処理内容を容易に設定可能とする効果が得られると共に、システムとしての汎用性向上の効果も得られる。

【0041】実施の形態4。図1において、49はウィンドウ整列手段であり、このウィンドウ整列手段49は図11(a)に示すグループリスト70、ウィンドウリスト76、整列位置データ77に基づき、オペレータが選択した整列条件に合致するウィンドウを識別し、システムまたはアプリケーションS/Wにより設定された整列位置にウィンドウを整列させるもので、ウィンドウ管理手段1には従来機能の他にウィンドウ整列手段49に必要な情報を通知する機能が付加されている。また、図2において、44はウィンドウをグルーピング（整列）化するための識別子（group_id）、45はグルーピングされたウィンドウ内での整列順序（priority）である。その他の構成については実施の形態1ないし3と同様なので、その重複する説明を省略する。

【0042】次に動作について説明する。図10はこの発明の実施の形態4によるウィンドウ整列手段49の動作を示すフローチャート、図11はウィンドウ整列手段49が管理するデータの構造と表示状態との対応関係を示す説明図である。実施の形態4におけるウィンドウ管理手段1は、自らの管理テーブルにウィンドウ制御データ30を登録する際に、ウィンドウをグルーピング化するための識別子（group_id）44を参照し、グルーピングを行なう値が設定されていた場合、登録通知として当該ウィンドウ制御データ30のウィンドウ識別子（win_id）41、ウィンドウをグルーピング化するための識別子（group_id）44、グルーピングされたウィンドウ内での整列順序（priority）45をウィンドウ整列手段49に通知し、ウィンドウが削除されると削除通知として当該ウィンドウ制御データ30のウィンドウ識別子（win_id）41、ウィンドウをグルーピング化するための識別子（group_id）44をウィンドウ整列手段49に通知する。また、マウスにより発生する特定の人カイベントを受信するとウィンドウ整列手段49に対し整列要求通知を発

行する。

【0043】図11(a)はウィンドウ整列手段49が作成し、ウィンドウ整列動作時に参照するデータを示している。グループリスト70はグルーピング時のグループを識別子であるグループID71、そのグループID71に対応するグループデータの先頭アドレス72により構成され、グループデータ75はそのグループに属するウィンドウデータ78のウィンドウリスト76と整列時の個々のウィンドウの座標データが格納されている整列位置データの先頭アドレス77により構成されている。また、整列位置データ83は、整列エリアの基準点のX座標（AREA_X）84、整列エリアの基準点のY座標（AREA_Y）85、整列エリアの幅（AREA_W）86、整列エリアの高さ（AREA_H）87、整列エリア内での整列方向（DIR）88、整列時のウィンドウの幅（RESIZE_W）89、整列時のウィンドウの高さ（RESIZE_H）90、整列方向に従った表示順序（POS_ID）91、整列時のウィンドウの基準点のX座標（POS_X）92、整列時のウィンドウの基準点のY座標（POS_Y）93の座標列により構成されている。

【0044】ウィンドウ整列手段49はシステム起動時に定義ファイルに設定されているグループID（整列条件）毎の整列エリア、整列時のウィンドウサイズ、整列方向を参照し、グループリスト70およびグループ毎のグループデータ75、整列位置データ83を作成して、グループリスト70のグループID71、整列位置データ83の整列エリアの基準点のX座標（AREA_X）84、整列エリアの基準点のY座標（AREA_Y）85、整列エリアの幅（AREA_W）86、整列エリアの高さ（AREA_H）87、整列エリア内での整列方向（DIR）88、整列時のウィンドウの幅（RESIZE_W）89、整列時のウィンドウの高さ（RESIZE_H）90に対応するデータを設定する。

【0045】これらのデータを設定し終わるとウィンドウ管理手段1は整列位置データ83の整列エリアの基準点のX座標（AREA_X）84、整列エリアの基準点のY座標（AREA_Y）85、整列エリアの幅（AREA_W）86、整列エリアの高さ（AREA_H）87、整列エリア内での整列方向（DIR）88、整列時のウィンドウの幅（RESIZE_W）89、整列時のウィンドウの高さ（RESIZE_H）90より整列エリア内を整列時のウィンドウサイズで分割した際の、個々のウィンドウの整列時のウィンドウの基準点のX座標（POS_X）92、整列時のウィンドウの基準点のY座標（POS_Y）93の座標列を算出し、配置場所を一意に定める整列方向に従った表示順序（POS_ID）91、整列時のウィンドウの基準点のX座標（POS_X）92、整列時のウィンドウの基準点のY座標（POS_Y）93を対応づけながら整列位置データ8

3に格納する（ステップST40）。

【0046】これら一連の整列位置データ算出の処理を終えるとウィンドウ整列手段49はウィンドウ管理手段1からの通知を待ち（ステップST41）、登録通知を受信すると（ステップST42）、当該ウィンドウのウィンドウデータ78を作成し、ウィンドウデータ78が表示順序80の小さい値順に並ぶよう当該グループのウィンドウリスト76に登録し、ウィンドウ管理手段1からの次の通知を待つ（ステップST43）。また削除通知を受信すると（ステップST44）、当該ウィンドウのウィンドウデータを当該グループのウィンドウリスト76より削除し、ウィンドウ管理手段1からの次の通知を待つ（ステップST45）。そして整列要求通知を受信すると（ステップST46）、整列条件（グループID71）選択用ポップアップウィンドウを表示し（ステップST47）、オペレータに入力を促す。

【0047】オペレータの操作によりポップアップウィンドウが閉じられるとウィンドウ整列手段49は入力されたデータを参照し、入力（選択）されたグループID71よりグループリスト70を検索し、当該グループのウィンドウリスト76、整列位置データ83の個々のデータを対応させ、整列処理を行なう。また何も入力されていなければウィンドウ管理手段1からの次の通知を待つ（ステップST48、ステップST49）。

【0048】以上のように、この実施の形態4によれば、オペレータに代わり、ウィンドウの表示内容を識別してグルーピング化を行ない、そのグループでの相対関係に応じた表示を行なうため、動画のコマ送り時の画面のように表示内容にほとんど差異がないようなものを規則的にあるいは関連づけて表示する際のオペレータの負荷を軽減する効果が得られる。

【0049】実施の形態5、図1において、50はウィンドウ整列手段49内に設けられ、実施の形態4ではシステム起動時に定義ファイルでしか設定できなかった整列位置データ83の整列エリアの基準点のX座標（AREA_X）84をオペレータがグループ毎に任意の値に設定可能とする整列時サイズ設定手段である。また、実施の形態5におけるウィンドウ管理手段1は、実施の形態4の機能の他に、マウスにより発生する特定の入力イベントを受信すると整列時サイズ設定手段50に対し整列時サイズ変更通知を発行する。その他の構成については実施の形態4と同様なので、その重複する説明を省略する。

【0050】次に動作について説明する。図12はこの発明の実施の形態5によるウィンドウ整列手段49の動作を示すフローチャートである。ウィンドウ管理手段1より整列時サイズ変更通知を受けると（ステップST51）、整列時サイズ設定手段50はデータ入力用のポップアップウィンドウ表示し、オペレータの入力を促す。オペレータの操作によりポップアップウィンドウが閉じ

られると整列時サイズ設定手段50は入力されたデータを参照し、入力（選択）されたグループID71よりグループリスト70を検索し、当該グループの整列位置データ83の整列時のウィンドウの幅（RESIZE_W）89、整列時のウィンドウの高さ（RESIZE_H）90に入力されたデータを設定し、実施の形態4の手順にて整列位置データ83を再計算し更新する（ステップST52）。また、何も入力されていなければウィンドウ管理手段1からの次の通知を待つ。なお、この実施の形態5におけるウィンドウ整列手段49のその他の処理は実施の形態4のウィンドウ整列手段49のものと同様である。

【0051】以上のように、この実施の形態5では、システム起動時にしか設定できなかった整列時のウィンドウサイズをシステム稼働中に動的に変更可能とすることで、表示内容および利用状況といった実情に応じたウィンドウサイズが設定可能となるため、表示画面の視認性および識別性の向上と共に、システムとしての汎用性向上の効果も得られる。

【0052】実施の形態6、図1において、51はウィンドウ整列手段49内に設けられ、実施の形態4、5ではシステム起動時に定義ファイルでしか設定できなかった整列位置データ83の整列エリア内での整列方向（DIR）88をオペレータがグループ毎に任意の値に設定可能とする整列方向選択手段である。また、実施の形態6におけるウィンドウ管理手段1は、実施の形態4、5の機能の他に、マウスにより発生する特定の入力イベントを受信すると整列方向選択手段51に対し整列方向選択通知を発行する。その他の構成については実施の形態4ないし5と同様なので、その重複する説明を省略する。

【0053】次に動作について説明する。図13はこの発明の実施の形態6によるウィンドウ整列手段49の動作を示すフローチャートである。ウィンドウ管理手段1より整列方向選択通知を受けると（ステップST53）、整列方向選択手段51はデータ入力用のポップアップウィンドウ表示し、オペレータの入力を促す。オペレータの操作によりポップアップウィンドウが閉じられると整列方向選択手段51は入力されたデータを参照し、入力（選択）されたグループID71よりグループリスト70を検索し、当該グループの整列位置データ83の整列エリア内での整列方向（DIR）88に入力されたデータを設定し、実施の形態4の手順にて整列位置データ83を再計算し更新する（ステップST52）。また、何も入力されていなければウィンドウ管理手段1からの次の通知を待つ。なお、この実施の形態6におけるウィンドウ整列手段49のその他の処理は実施の形態5のウィンドウ整列手段49のものと同様である。

【0054】以上のように、この実施の形態6では、実施の形態4、5ではシステム起動時にしか設定できな

った整列方向をシステム稼働中に動的に変更可能とすることで、表示内容および利用状況といった実情に応じたウィンドウレイアウトが設定可能となり、表示画面の視認性および識別性の向上と共に、システムとしての汎用性向上の効果も得られる。

【0055】実施の形態7、図1において、52はウィンドウ整列手段49内に設けられ、実施の形態4～6ではシステム起動時に定義ファイルでしか設定できなかった整列位置データ83の整列エリアの基準点のX座標(AREA_X)84、整列エリアの基準点のY座標(AREA_Y)85、整列エリアの幅(AREA_W)86、整列エリアの高さ(AREA_H)87をオペレータがグループ毎に任意の値に設定可能とする整列エリア設定手段である。また、実施の形態7におけるウィンドウ管理手段1は、実施の形態4～6の機能の他に、マウスにより発生する特定の入力イベントを受信すると整列エリア設定手段52に対し整列エリア設定通知を発行する。その他の構成については実施の形態4～6と同様なので、その重複する説明を省略する。

【0056】次に動作について説明する。図14はこの発明の実施の形態7によるウィンドウ整列手段49の動作を示すフローチャートである。ウィンドウ管理手段1より整列エリア設定通知を受けると(ステップST54)、整列エリア設定手段52はデータ入力用のポップアップウィンドウ表示し、オペレータの入力を促す。オペレータの操作によりポップアップウィンドウが閉じられると整列エリア設定手段52は入力されたデータを参照し、入力(選択)されたグループID71よりグループリスト70を検索し、当該グループの整列位置データ83の整列エリアの基準点のX座標(AREA_X)84、整列エリアの基準点のY座標(AREA_Y)85、整列エリアの幅(AREA_W)86、整列エリアの高さ(AREA_H)87に入力されたデータを設定し、実施の形態4の手順にて整列位置データ83を再計算し更新する(ステップST52)。また、何も入力されていなければウィンドウ管理手段1からの次の通知を待つ。なお、この実施の形態7におけるウィンドウ整列手段49のその他の処理は実施の形態6のウィンドウ整列手段49のものと同様である。

【0057】以上のように、この実施の形態7では、実施の形態4～6ではシステム起動時にしか設定できなかった整列エリアをシステム稼働中に動的に変更可能とすることで、表示内容および利用状況といった実情に応じたウィンドウレイアウトがより細やかに設定可能となり、表示画面の視認性および識別性の向上と共に、システムとしての汎用性向上の効果も得られる。

【0058】実施の形態8、図1において、53はウィンドウ管理手段1内に設けられ、実施の形態4～7ではアプリケーションS/Wがウィンドウ生成時にしか設定できなかったウィンドウ管理手段1が管理するウィンド

ウ制御データ30のウィンドウをグルーピング化するための識別子(group_id)44、グルーピングされたウィンドウ内での整列順序(priority)45を任意の値に設定可能とする整列用属性設定手段である。その他の構成については実施の形態4～7と同様なので、その重複する説明を省略する。

【0059】次に動作について説明する。図15はこの発明の実施の形態8によるウィンドウ整列手段49の動作を示すフローチャートである。整列用属性設定手段53はウィンドウのタイトルバー等に用意されている整列用属性変更ボタンをマウス等のポインティングデバイスによりクリックするとデータ入力用のポップアップウィンドウ表示し、オペレータの入力を促す。オペレータの操作によりポップアップウィンドウが閉じられると整列用属性設定手段53は入力されたデータを参照し、当該ウィンドウ制御データ30のウィンドウをグルーピング化するための識別子(group_id)44、グルーピングされたウィンドウ内での整列順序(priority)45を入力されたデータで更新すると同時に、整列用属性通知として当該ウィンドウ制御データ30のウィンドウ識別子(win_id)41、ウィンドウをグルーピング化するための識別子(group_id)44、グルーピングされたウィンドウ内での整列順序(priority)45をウィンドウ整列手段49に通知する(ステップST56、ステップST57)。整列用属性変更通知を受信したウィンドウ整列手段49は通知されたウィンドウをグルーピング化するための識別子(group_id)44の値により当該ウィンドウデータをウィンドウリスト75より削除したり、登録したり、また場合によってはグループリスト70へのグループデータの登録/削除も行なう。なお、この実施の形態8におけるウィンドウ整列手段49のその他の処理は実施の形態7のウィンドウ整列手段49のものと同様である。

【0060】以上のように、この実施の形態8では、実施の形態4～7ではシステム起動時にしか設定できなかった整列条件をシステム稼働中に動的に設定可能とすることで、表示内容および利用状況といった実情に応じたウィンドウのグルーピングおよび配置が可能になり、表示画面の視認性および識別性の向上と共に、システムとしての汎用性向上の効果も得られる。

【0061】

【発明の効果】以上のように、請求項1記載の発明によれば、設定された基準時間を通じてウィンドウが利用されなかった場合に該当するウィンドウを非表示処理する自動非表示処理手段を備えるように構成したので、自動非表示処理手段はオペレータに代わりウィンドウの利用度を識別でき、利用度の低いウィンドウを自動的に非表示とすることで、表示内容確認のためのオペレータの操作を軽減できると共に、同時に監視画面の視認性および

識別性を向上することができる効果がある。

【0062】請求項2記載の発明によれば、ウィンドウ毎に基準時間を任意に設定する非表示判定値設定手段を備えるように構成したので、システム稼働中に変更できなかった基準時間の値をシステム稼働中でも変更可能とすることで、表示内容および利用状況といった実情に応じた判定値が柔軟に設定可能となるため、容易に最適値を設定可能とする効果が得られると共にシステムとしての汎用性を向上させる効果がある。

【0063】請求項3記載の発明によれば、ウィンドウ毎に非表示処理内容をアイコン化またはウィンドウ削除に任意に設定する非表示処理選択手段を備えるように構成したので、システム稼働中に変更できなかった非表示処理内容の値をシステム稼働中でも変更可能とすることで、表示内容および利用状況といった実情に応じた非表示処理が柔軟に設定可能となるため、最適な非表示処理内容を容易に設定可能とすることができると共に、システムとしての汎用性を向上させる効果がある。

【0064】請求項4記載の発明によれば、ウィンドウの整列時の識別子および整列時のウィンドウの整列順序に応じて定義ファイルに設定された整列時のウィンドウサイズ、整列エリアの座標および整列アルゴリズムに基づいて整列時の各ウィンドウの表示位置を演算するウィンドウ整列手段を備えるように構成したので、オペレータに代わり、ウィンドウの表示内容を識別して整列でき、そのグループでの相対関係に応じた表示を行なうため、動画のコマ送り時の画面のように表示内容にほとんど差異がないようなものを規則的にあるいは関連づけて表示する際のオペレータの負荷を軽減できる効果がある。

【0065】請求項5記載の発明によれば、整列時のウィンドウサイズを任意に設定する整列時サイズ設定手段を備えるように構成したので、システム起動時にしか設定できなかった整列時のウィンドウサイズをシステム稼働中に動的に変更可能とすることで、表示内容および利用状況といった実情に応じたウィンドウサイズが設定可能となるため、表示画面の視認性および識別性を向上させると共に、システムとしての汎用性を向上させる効果がある。

【0066】請求項6記載の発明によれば、整列アルゴリズムを任意に設定する整列方向選択手段を備えるように構成したので、システム起動時にしか設定できなかった整列方向をシステム稼働中に動的に変更可能とすることで、表示内容および利用状況といった実情に応じたウィンドウレイアウトが設定可能となり、表示画面の視認性および識別性を向上させると共に、システムとしての汎用性を向上させる効果がある。

【0067】請求項7記載の発明によれば、整列エリアの座標を任意に設定する整列エリア設定手段を備えるように構成したので、システム起動時にしか設定できな

かった整列エリアをシステム稼働中に動的に変更可能とすることで、表示内容および利用状況といった実情に応じたウィンドウレイアウトがより細やかに設定可能となり、表示画面の視認性および識別性を向上させると共に、システムとしての汎用性を向上させる効果がある。

【0068】請求項8記載の発明によれば、ウィンドウの整列時の識別子および整列時のウィンドウの整列順序を任意に設定する整列用属性設定手段を備えるように構成したので、システム起動時にしか設定できなかった整列条件をシステム稼働中に動的に設定可能とすることで、表示内容および利用状況といった実情に応じたウィンドウのグルーピングおよび配置が可能になり、表示画面の視認性および識別性を向上させると共に、システムとしての汎用性を向上させる効果がある。

【図面の簡単な説明】

【図1】 この発明の実施の形態によるマルチウィンドウ表示装置を示す機能ブロック図である。

【図2】 この発明の実施の形態によるウィンドウ管理手段が管理するウィンドウ管理テーブルを示す概念図である。

【図3】 自動非表示処理手段の動作を示すフローチャートである。

【図4】 自動非表示処理手段が管理する非表示ウィンドウデータ、および非表示対象ウィンドウリストを示す概念図である。

【図5】 非表示ウィンドウデータの遷移を示す概念図である。

【図6】 この発明の実施の形態2による非表示判定値設定手段が表示するポップアップウィンドウとその動作を示すフローチャートである。

【図7】 この発明の実施の形態2における自動非表示処理手段の動作を示すフローチャートである。

【図8】 この発明の実施の形態3による自動非表示処理手段の動作を示すフローチャートである。

【図9】 この発明の実施の形態3による非表示処理選択手段が表示するポップアップウィンドウとその動作を示すフローチャートである。

【図10】 この発明の実施の形態4によるウィンドウ整列手段の動作を示すフローチャートである。

【図11】 ウィンドウ整列手段が管理するデータの構造と表示状態との対応関係を示す説明図である。

【図12】 この発明の実施の形態5によるウィンドウ整列手段の動作を示すフローチャートである。

【図13】 この発明の実施の形態6によるウィンドウ整列手段の動作を示すフローチャートである。

【図14】 この発明の実施の形態7によるウィンドウ整列手段の動作を示すフローチャートである。

【図15】 この発明の実施の形態8によるウィンドウ整列手段の動作を示すフローチャートである。

【図16】 従来のマルチウィンドウ表示装置を示す機

能ブロック図である。

【図17】 マルチウィンドウ表示装置のシステム構成図である。

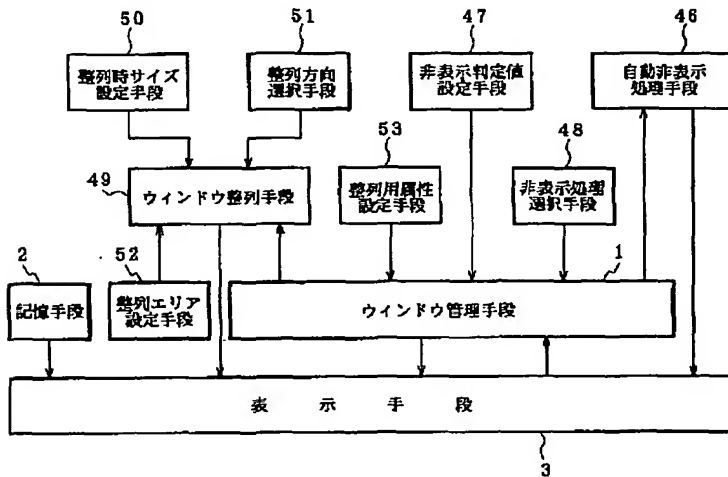
【図18】 ウィンドウ管理テーブルに生成される格納情報を示す概念図である。

【図19】 マイクロプロセッサにより2つのウィンドウを生成した時のウィンドウメモリおよびウィンドウ管理テーブルの状態を示す説明図である。

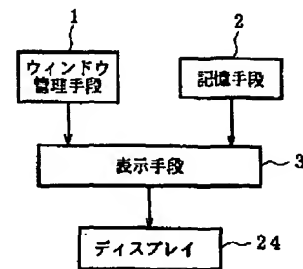
【符号の説明】

1 ウィンドウ管理手段、2 記憶手段、3 表示手段、30 ウィンドウ制御データ、46 自動非表示処理手段、47 非表示判定値設定手段、48 非表示処理選択手段、49 ウィンドウ整列手段、50 整列時サイズ設定手段、51 整列方向選択手段、52 整列エリア設定手段、53 整列用属性設定手段。

【図1】



【図16】



【図2】

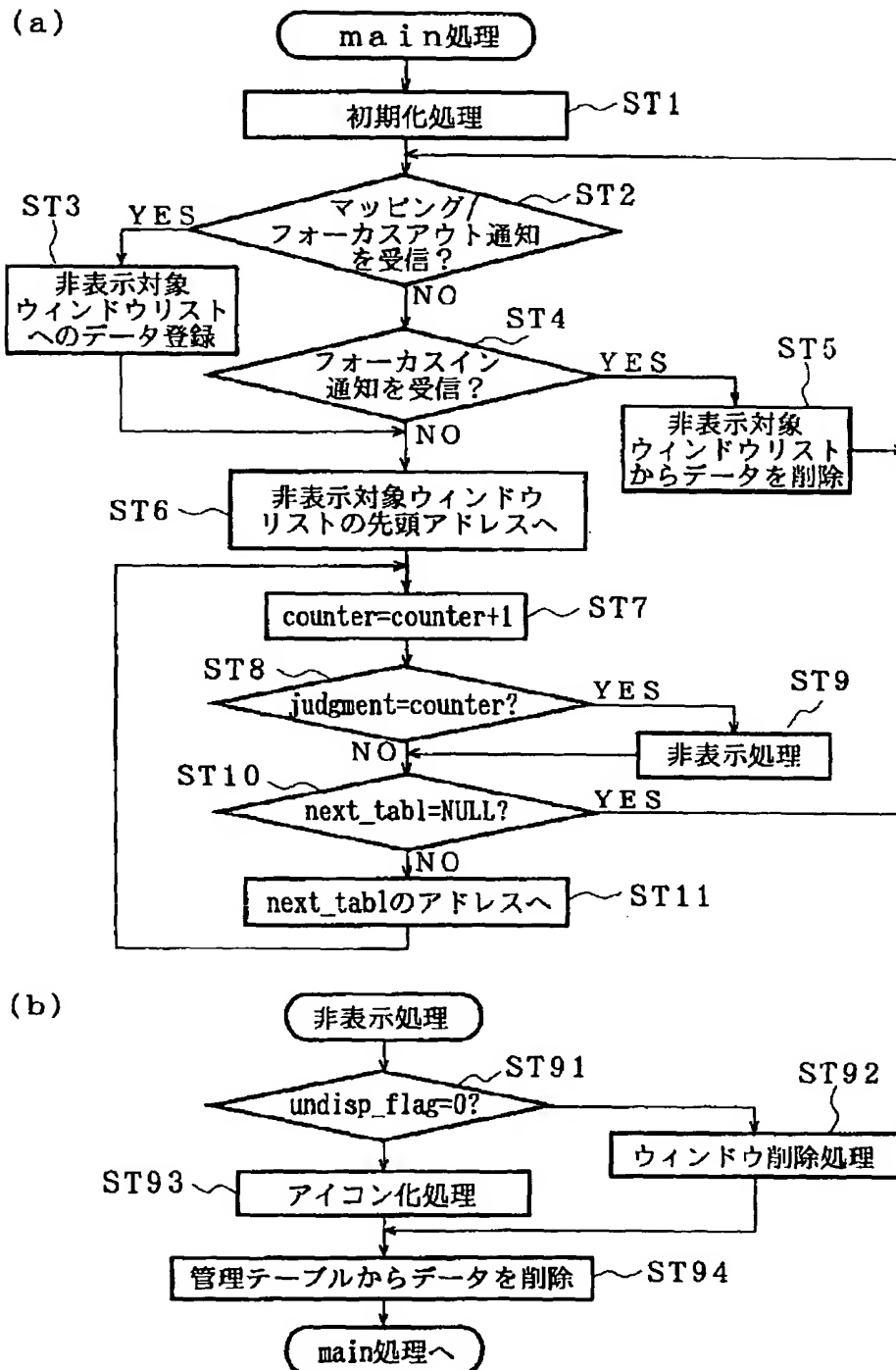
30		
31	LBW	・・・論理バッファの幅
32	LBH	・・・論理バッファの高さ
33	WIW	・・・ウィンドウの幅
34	WIH	・・・ウィンドウの高さ
35	WIX	・・・ウィンドウの基点のX座標
36	WIY	・・・ウィンドウの基点のY座標
37	FP	・・・表示優先順位が次のウィンドウへのポインタ
38	BP	・・・表示優先順位が前のウィンドウへのポインタ
39	FGC	・・・ウィンドウの前景色
40	BGC	・・・ウィンドウの背景色
41	win_id	・・・ウィンドウ識別子
42	judge	・・・非表示と判定するカウント数
43	process	・・・非表示処理の内容
44	group_id	・・・グルーピング時の識別子
45	priority	・・・グループ内での整列順序

【図18】

30		
31	LBW	・・・論理バッファLBPFの幅
32	LBH	・・・論理バッファLBPFの高さ
33	WIW	・・・ウィンドウの幅
34	WIH	・・・ウィンドウの高さ
35	WIX	・・・ウィンドウの起点(左上点)のX座標
36	WIY	・・・ウィンドウの起点(左上点)のY座標
37	FP	・・・次のウィンドウへのポインタ
38	BP	・・・前のウィンドウへのポインタ
39	FGC	・・・ウィンドウの前景色
40	BGC	・・・ウィンドウの背景色

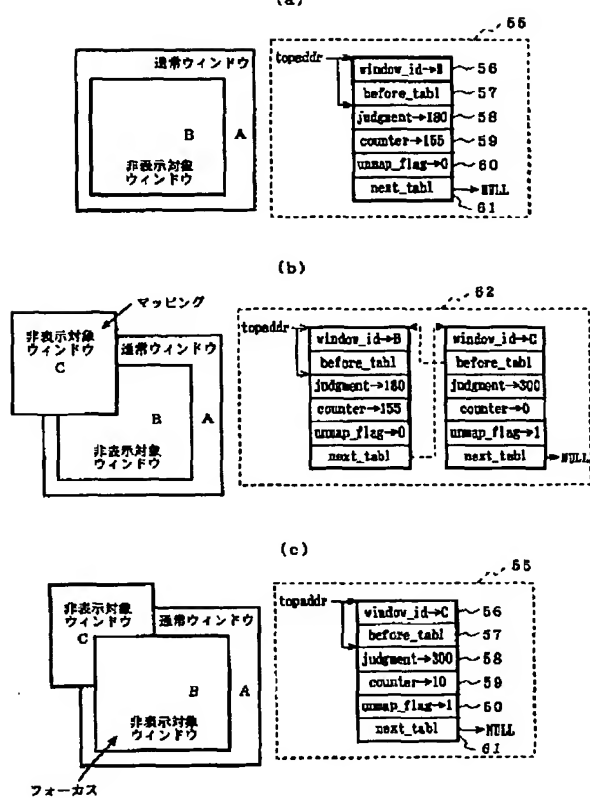
30: ウィンドウ制御データ

【図3】

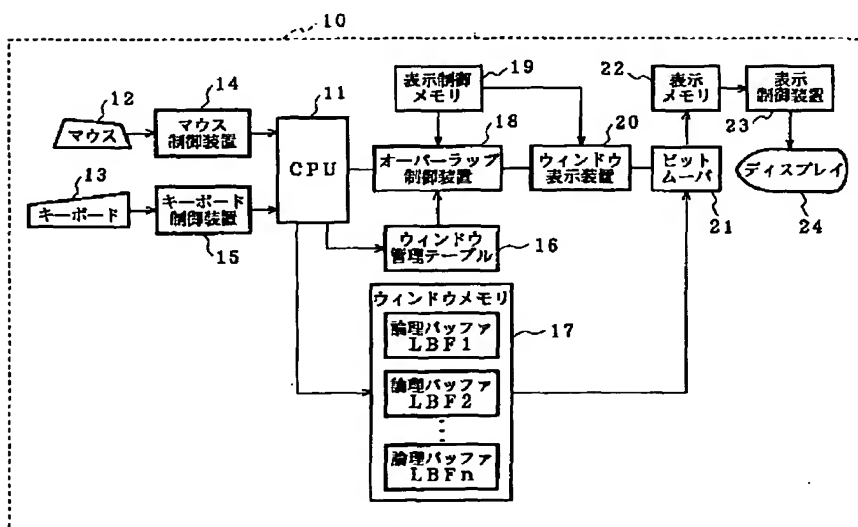


【図 5】

(a)

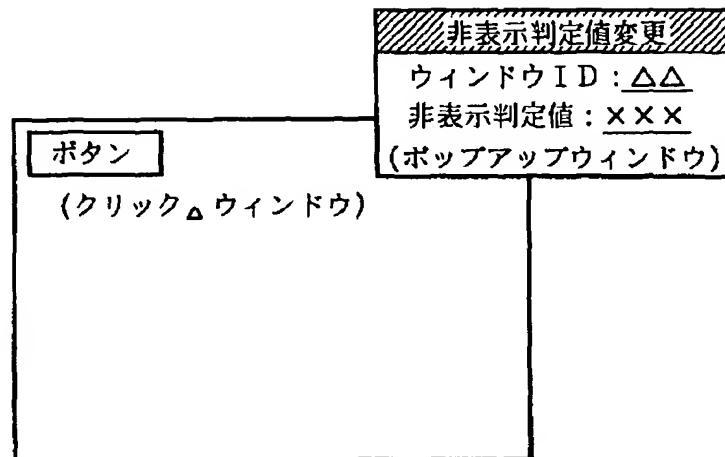


【図 17】

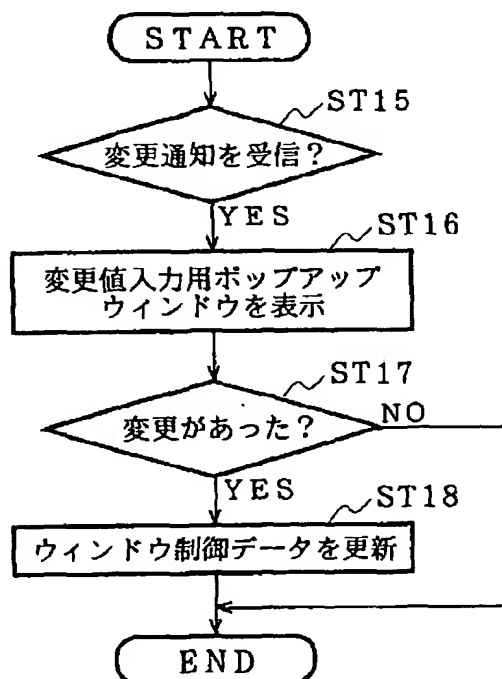


【図6】

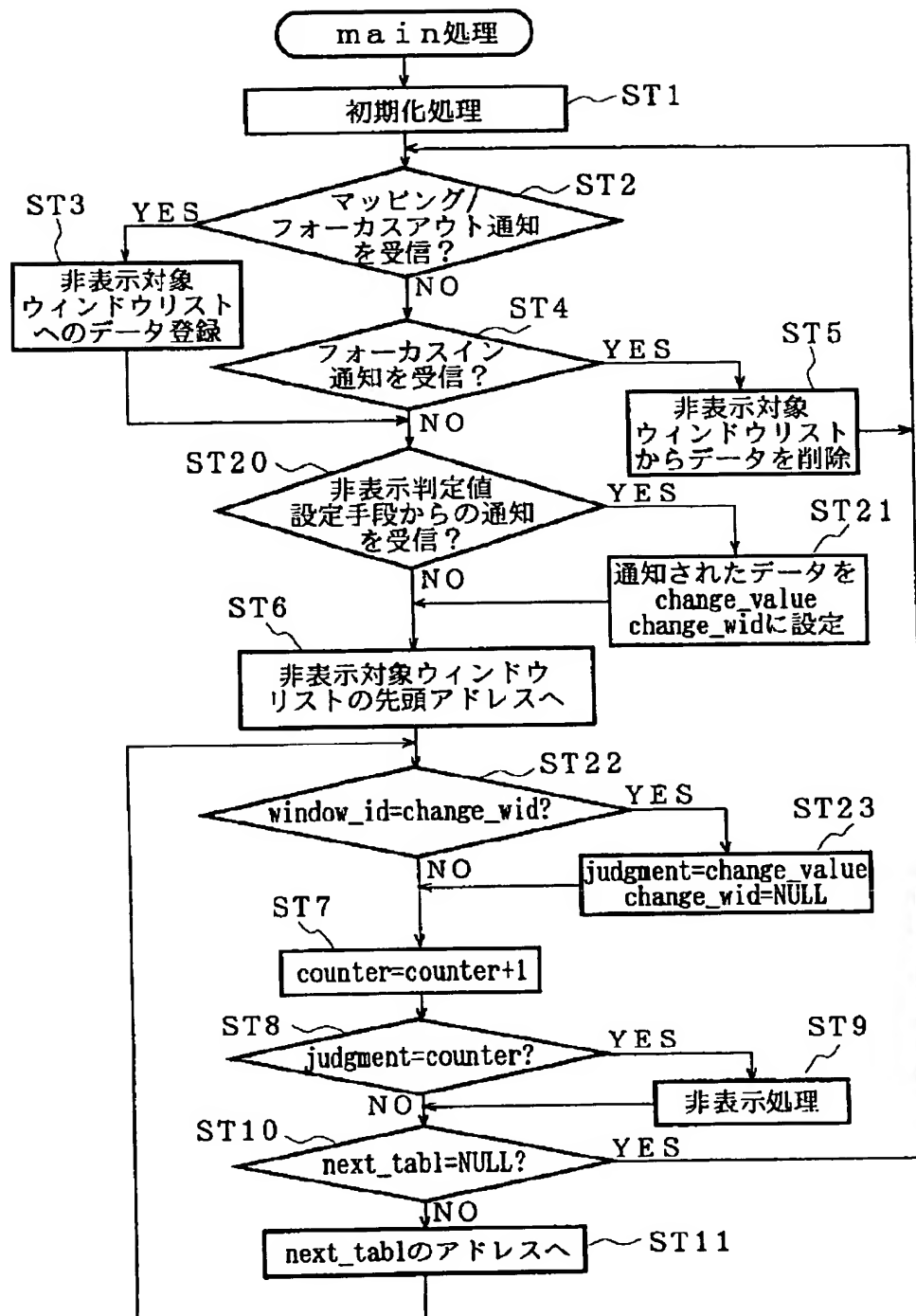
(a)



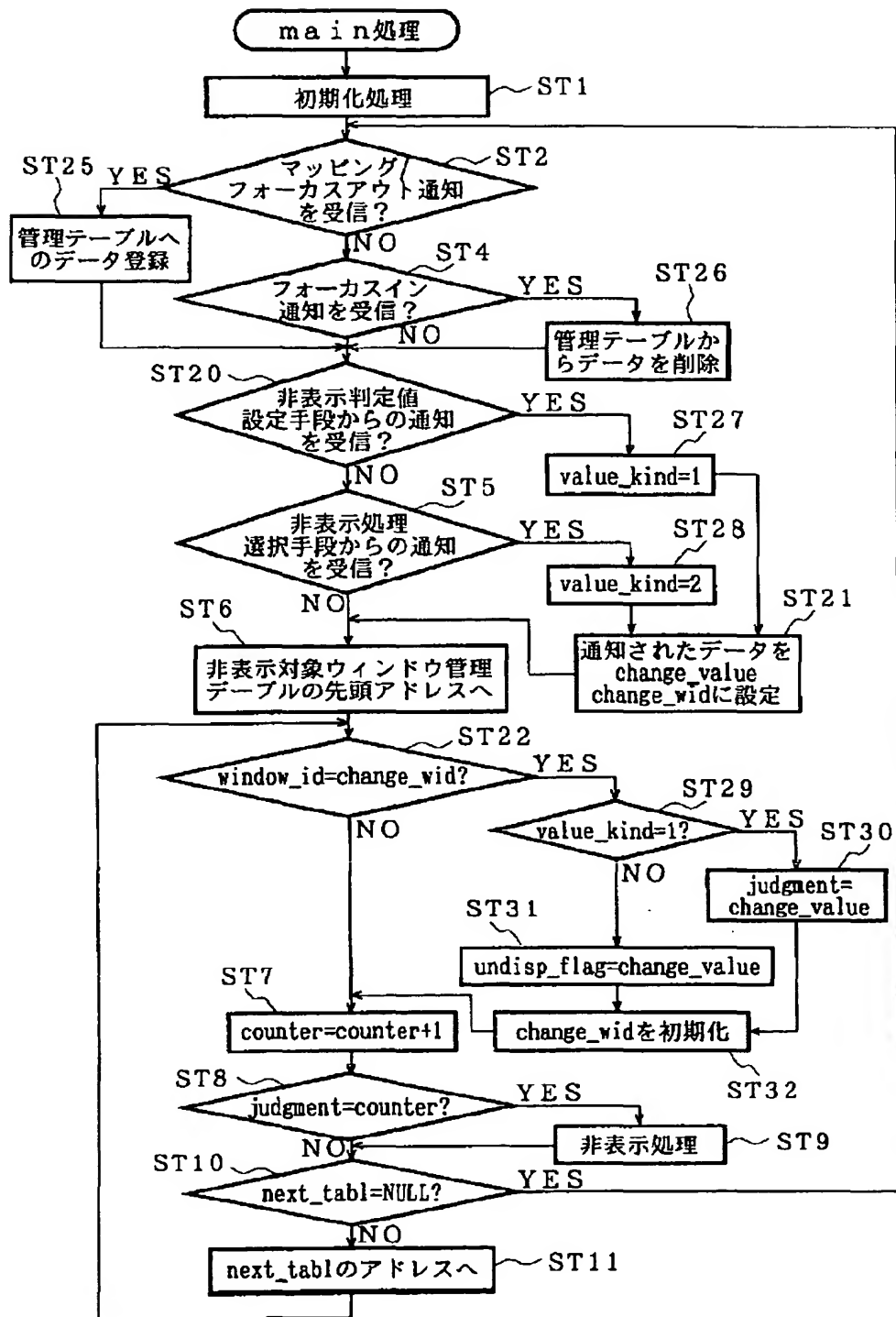
(b)



【図7】

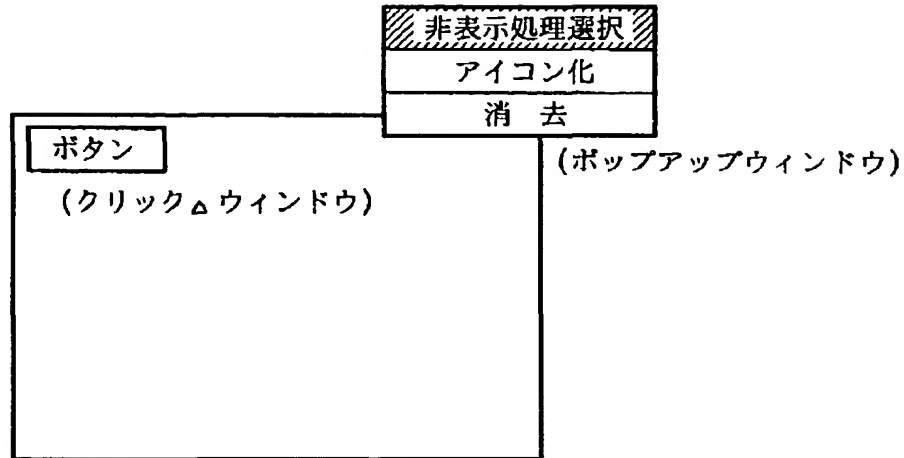


【図8】

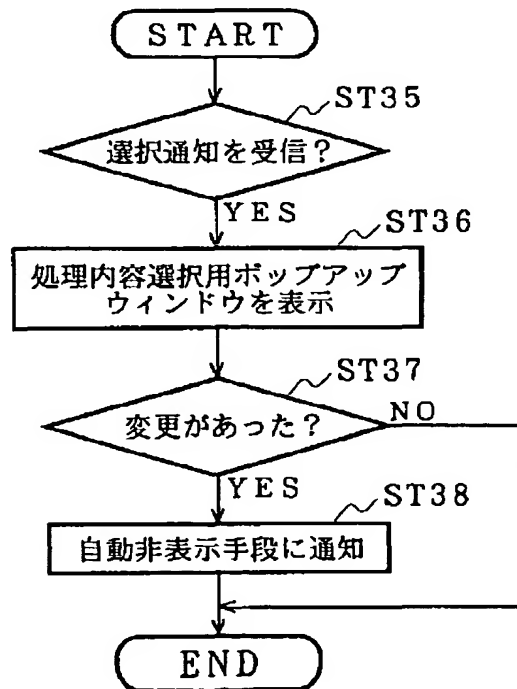


【図 9】

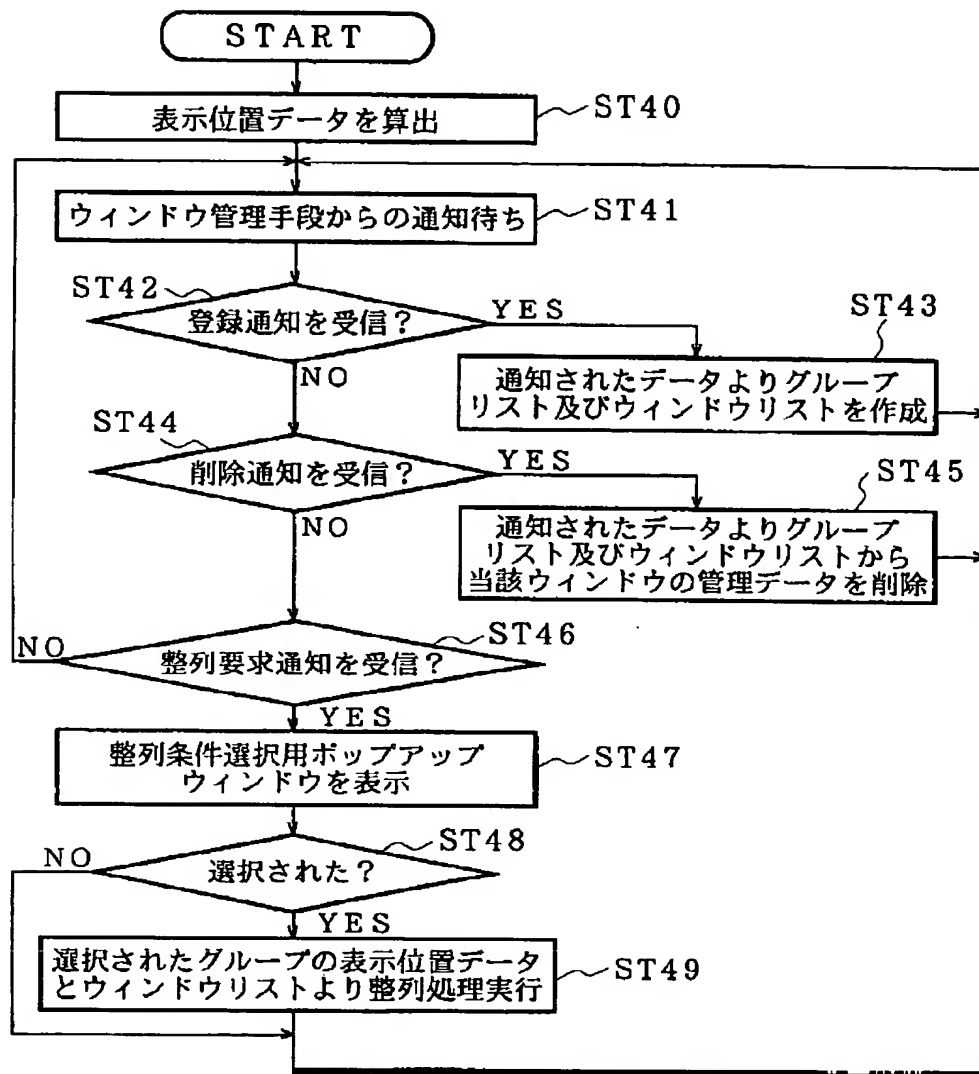
(a)



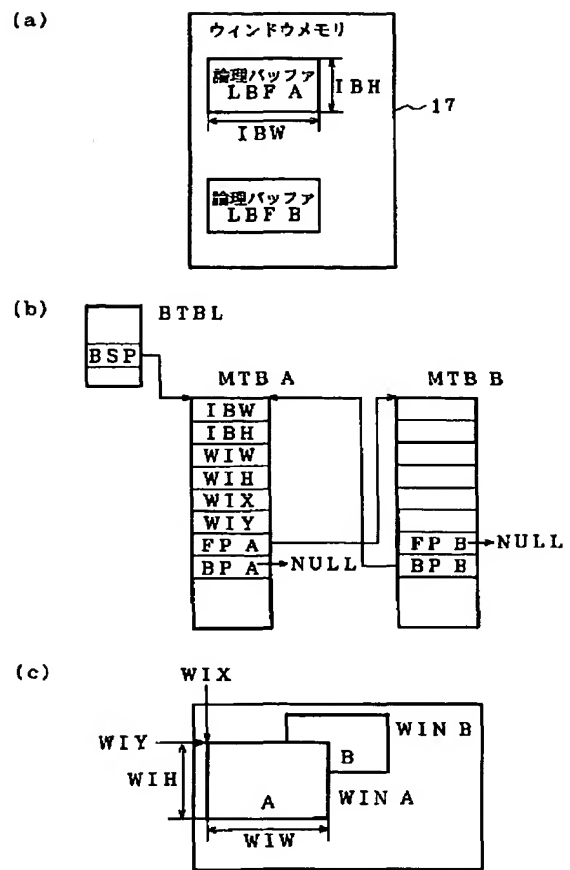
(b)



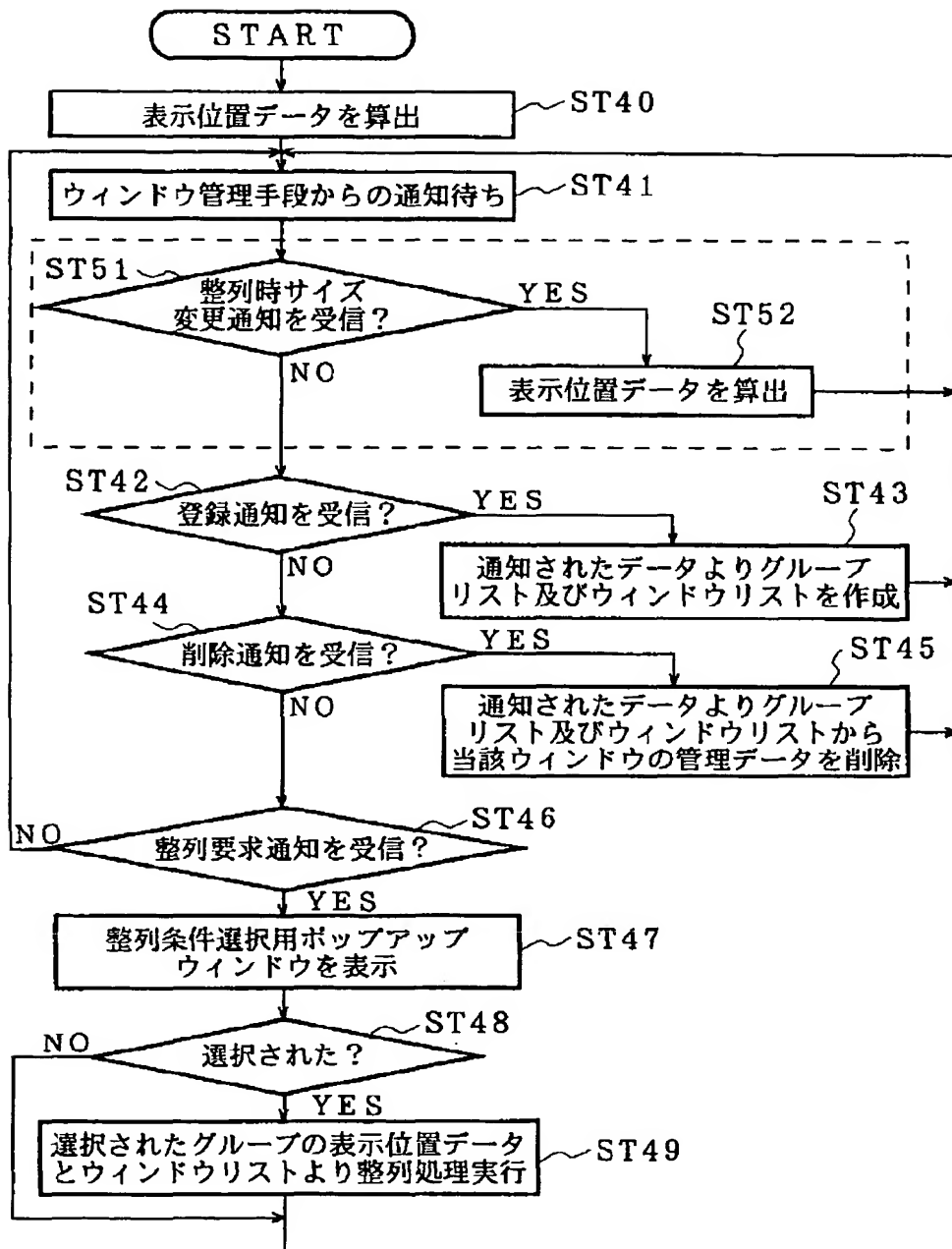
【図10】



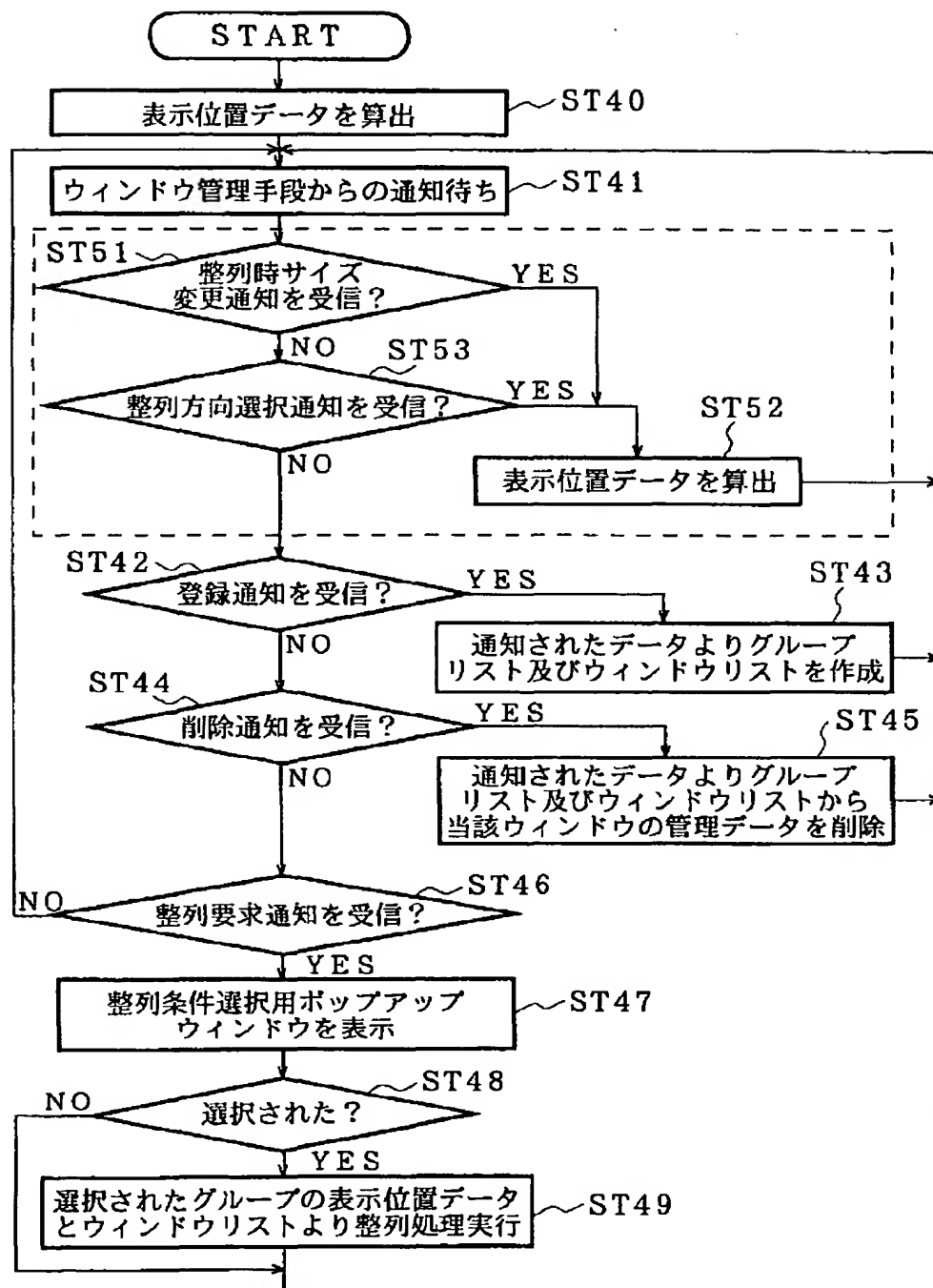
【図 19】



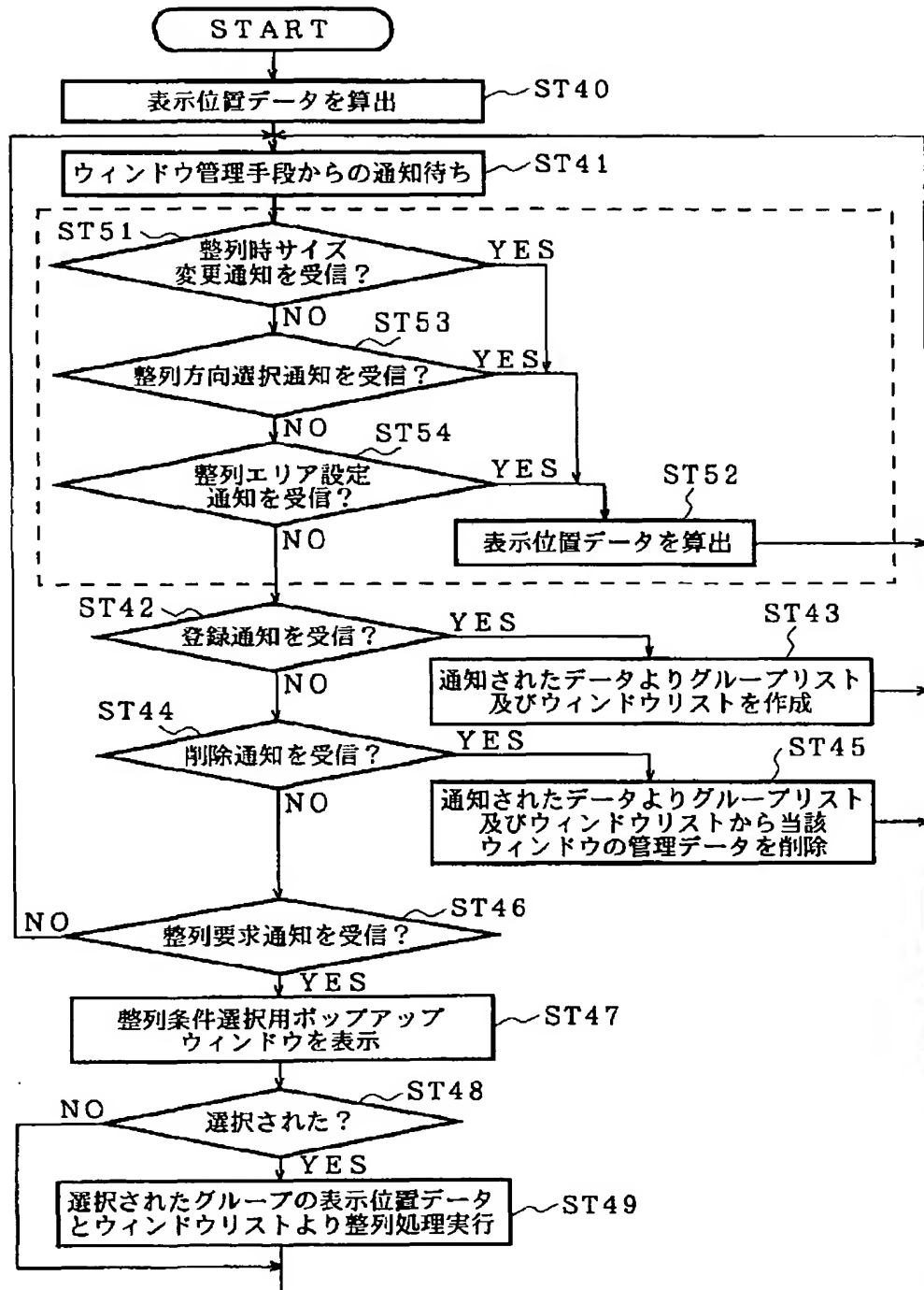
【図12】



【図13】



【図14】



【図15】

